

JP2004252665

Publication Title:

DOCUMENT PROCESSING METHOD

Abstract:

Abstract of JP2004252665

PROBLEM TO BE SOLVED: To produce a template of a document file on the basis of an existing document.

SOLUTION: When selecting a 'setting of template' item from an editing window of the document, a 'registration of present setting' button is displayed together with a selectable icon of template information. When the 'registration of present setting' button is selected, data on a book attribute of the document during editing are read (S2201), and set contents are displayed (S2201). In response thereto, when a template name is inputted and when an 'OK' button is pushed, present setting is registered as the template (S2205).

COPYRIGHT: (C)2004,JPO&NCIP

Data supplied from the esp@cenet database - Worldwide

Courtesy of <http://v3.espacenet.com>

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2004-252665

(P2004-252665A)

(43) 公開日 平成16年9月9日(2004.9.9)

(51) Int. Cl.⁷

G06F 17/21

G06F 3/12

F I

G06F 17/21

G06F 3/12

566E

F

テーマコード(参考)

5B009

5B021

審査請求 未請求 請求項の数 1 O L (全 25 頁)

(21) 出願番号 特願2003-41544 (P2003-41544)
(22) 出願日 平成15年2月19日(2003.2.19)

(71) 出願人 000001007
キヤノン株式会社
東京都大田区下丸子3丁目30番2号
(74) 代理人 100076428
弁理士 大塚 康德
(74) 代理人 100112508
弁理士 高柳 司郎
(74) 代理人 100115071
弁理士 大塚 康弘
(74) 代理人 100116894
弁理士 木村 秀二
(72) 発明者 森 安生
東京都大田区下丸子3丁目30番2号 キ
ヤノン株式会社内

最終頁に続く

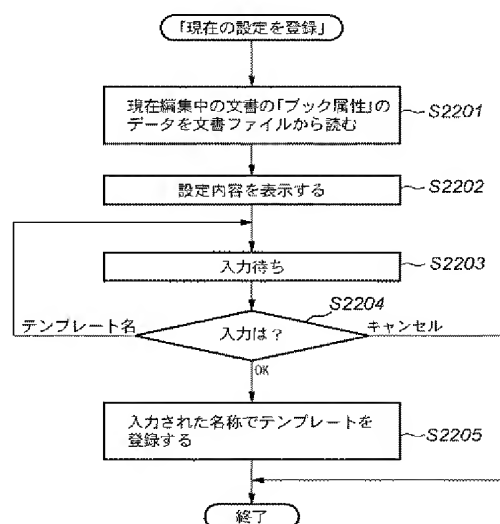
(54) 【発明の名称】 文書処理方法

(57) 【要約】

【課題】文書ファイルのテンプレートを既存の文書を基に作成する。

【解決手段】文書の編集ウインドウから「テンプレートを設定」項目を選択すると、選択できるテンプレート情報のアイコンと共に、「現在の設定を登録」ボタンが表示される。「現在の設定を登録」ボタンが選択されると、編集中の文書のブック属性のデータが読まれ(S2201)、設定内容が表示される(S2201)。それに対してテンプレート名称が入力され、「OK」ボタンが押されると、現在の設定がテンプレートとして登録される(S2205)。

【選択図】図21



【特許請求の範囲】**【請求項1】**

文書を印刷する際の印刷設定を変更可能な文書処理方法であって、
編集中の文書の印刷設定をテンプレート情報として登録する登録工程と、
前記登録工程により登録されたテンプレート情報を、編集中の文書または新規な文書の印刷設定として読み込む読み込み工程と
を備えることを特徴とする文書処理方法。

【発明の詳細な説明】**【0001】****【発明の属する技術分野】**

本発明は、例えば文書処理プログラムによって生成される文書データに対する印刷設定機能を提供する文書処理方法に関するものである。

【0002】**【従来の技術】**

文字や表、画像など、データの種類が異なると、それらデータを定義する構造やそれらデータに対する編集操作が異なるために、各種データの種類の応じた様々なアプリケーションプログラムが提供されている。従って、これらのデータを利用する利用者は、文字を編集するためには文字処理プログラム、表を編集するためには表計算プログラム、画像を編集するためには画像編集プログラムといった具合に、データの種類別にアプリケーションを使い分ける必要がある。

【0003】

このように、利用者はデータの種類ごとに、それに応じたアプリケーションプログラムを使い分けるのが一般的である。しかし、利用者が作成しようとする文書としては、例えば文字のみや、表のみ、画像のみといった1種類のデータのみで構成される文書よりも、文字と表、文字と画像などというように、複数種類のデータから構成される文書を扱う方が一般的である。そこで複数種類のデータを含む文書を作成するために、利用者は各種アプリケーションが備えている印刷機能を利用して、各アプリケーションを用いて各データを印刷させ、その印刷した印刷物を所望の順序に組み合わせて、所望の文書を作成する必要があった。

【0004】

或いは、例えばオフィススイートと呼ばれる、各種アプリケーションによって一つの統合アプリケーションを形成するプログラムによれば、各アプリケーションで生成されたデータ同士を組み合わせる一つの文書を構成する機能が提供される。この統合アプリケーションを使用することにより、利用者は、各アプリケーションで作成したデータを、その統合アプリケーションに含まれる特定のアプリケーションによって一つの文書にまとめることができる。

【0005】

従来このような統合アプリケーションでは、編集された文書に対して設定した、ページレイアウト、出力用紙、片面／両面／製本などの情報をテンプレートとしてあらかじめクライアントに登録しておく機能が提供されている。そしてユーザは、文書の新規作成時には登録されている何パターンかのテンプレートの中から所望の設定のテンプレートを選択することによって、その文書ファイルの編集を容易に行うことを可能としている。

【0006】

このテンプレートはファイルに変換して保存することも可能で、ユーザはテンプレートファイルを新規に作成することや、既存のテンプレートデータを変更することも可能である。またこのようなテンプレートファイルを他のPCへ持ち出すことや、または他のPCからテンプレートデータを取り込むことを可能とする、テンプレートファイルのインポート／エクスポート機能なども提供されている。

【0007】**【発明が解決しようとする課題】**

しかしながら、従来のシステムでは利用者はこのようなテンプレートを作成するには、アプリケーションで用意されている既存のテンプレートの設定を操作することで自分で好みの属性値に変えるか、またはデフォルト値（例えば、1UP、片面、フィニッシングなしのような設定）を初期値として新規にテンプレートを作成し、それを登録するという方法しかなかった。しかし、実際にはユーザは文書ファイルに対してあらかじめ必要な設定を済ませている場合が多い。そのため既存の文書ファイルの設定をテンプレート化したい場合などには、テンプレートを作成するために、デフォルトの初期値から再度テンプレートについて文書ファイルについてと同一ような編集設定作業を行う必要があった。

【0008】

本発明は上記従来例に鑑みてなされたもので、文書ファイルですでに設定されている出力用紙、ページレイアウト、製本などの設定情報をもとにテンプレートを作成することができる文書処理方法を提供することを目的とする。

【0009】

【課題を解決するための手段】

上記目的を達成するために本発明の文書処理装置は以下のような構成を備える。即ち、現在編集中の文書の設定情報をもとに、クライアントPC上に新しいテンプレートを登録する手段を有することを特徴とする。

【0010】

【発明の実施の形態】

以下、添付図面を参照して本発明の好適な実施の形態を詳細に説明する。

【0011】

＜システム概要＞

本発明の実施の形態1に係る文書処理システムの概要を、図1～図12を参照して説明する。この文書処理システムでは、一般的なアプリケーションにより作成されたデータファイルが、電子原稿ライタによって電子原稿ファイルに変換される。製本アプリケーションは、その電子原稿ファイルを編集する機能を提供している。以下、その詳細を説明する。

【0012】

＜システム構成及び動作＞

図1は、本発明の実施の形態に係る情報処理システムに好適な文書処理システムのソフトウェア構成を示す図である。

【0013】

この文書処理システムは、本実施の形態に係る情報処理装置の好適な実施形態であるデジタルコンピュータ100によって実現されている。一般アプリケーション101は、ワードプロセッシングやスプレッドシート、フォトレタッチ、ドロー或いはペイント、プレゼンテーション、テキスト編集などの機能を提供するアプリケーションプログラムであり、印刷機能を有している。これらアプリケーションプログラムは、作成された文書データや画像データなどのアプリケーションデータを印刷するにあたって、オペレーティングシステム（OS）により提供される所定のインターフェースを利用する。即ち、アプリケーション101は、作成したデータを印刷するために、前記インターフェースを提供するOSの出力モジュールに対して、予め定められる、OSに依存する形式の出力コマンド（GDI関数）を送信する。この出力コマンドを受けた出力モジュールは、プリンタ等の出力デバイスが処理可能な形式にそのコマンドを変換し、その変換されたコマンド（DDI関数）を出力する。この出力デバイスが処理可能な形式は、デバイスの種類やメーカ、機種などによって異なるために、デバイスごとにデバイスドライバが提供されており、OSでは、そのデバイスドライバを利用してコマンドの変換を行うことにより印刷データを生成し、JL（Job Language）でくくることにより印刷ジョブを生成する。このOSとして、例えばマイクロソフト社のウインドウズ（登録商標）を利用する場合は、前述した出力モジュールとしてはGDI（Graphic Device Interface）と呼ばれるモジュールが相当する。

【0014】

電子原稿ライタ１０２は、前述のデバイスドライバを改良したものであり、この文書処理システム実現のために提供されるソフトウェアモジュールである。但し、この電子原稿ライタ１０２は、特定の出力デバイスを目的としておらず、後述の製本アプリケーション１０４やプリンタドライバ１０６により処理可能な形式に出力コマンドを変換することにより電子原稿ファイル１０３を生成する。この電子原稿ライタ１０２による変換後の形式（以後、「電子原稿形式」と呼ぶ）は、ページ単位の原稿を詳細な書式をもって表現可能であれば特に問わない。実質的な標準形式のうちでは、例えばアドビシステムズによるPDF形式や、SGML形式などが電子原稿形式として採用できる。アプリケーション１０１により電子原稿ライタ１０２を利用させる場合には、出力に使用するデバイスドライバとして電子原稿ライタ１０２を指定してから印刷を実行させる。但し、電子原稿ライタ１０２によって作成されたままの電子原稿ファイルは、電子原稿ファイルとして必要とされる全ての設定が施されているとは限らない。そのため、デバイスドライバとして電子原稿ライタ１０２を指定するのは製本アプリケーション１０４であり、その管理下でアプリケーションデータの電子原稿ファイルへの変換が実行される。製本アプリケーション１０４は、電子原稿ライタ１０２が生成した新規の電子原稿ファイルを、後述する形式を備えた電子原稿ファイルとして完成させる。以下では、この点を明瞭に識別する必要がある際には、電子原稿ライタ１０２によって作成されたファイルを「電子原稿ファイル」と呼び、製本アプリケーション１０４によって構造を与えられた電子原稿ファイルを「ブックファイル」と呼ぶ。また、特に区別する必要がない場合は、アプリケーションにより生成されるドキュメントファイル、電子原稿ファイル、及びブックファイルをいずれも文書ファイル（または文書データ）と呼ぶ。

【００１５】

このようにデバイスドライバとして電子原稿ライタ１０２を指定し、一般アプリケーション１０１によりそのデータを印刷させることで、アプリケーションデータはアプリケーション１０１によって定義されたページ（以後、「論理ページ」或いは「原稿ページ」と呼ぶ）を単位とする電子原稿形式に変換され、電子原稿ファイル１０３としてハードディスクなどの記憶媒体に格納される。尚、ハードディスクは、本実施形態に係る文書処理システムを実現するコンピュータが備えているローカルドライブであってもよいし、ネットワークに接続されている場合にはネットワーク上に提供されるドライブであっても良い。

【００１６】

製本アプリケーション１０４は、「電子原稿ファイル」或いは「ブックファイル」１０３を読み込み、それを編集するための機能を利用者に提供する。但し、製本アプリケーション１０４は、各ページの内容を編集する機能は提供しておらず、ページを最小単位として構成される、後述する章やブックの構造を編集するための機能を提供している。

【００１７】

製本アプリケーション１０４によって編集されたブックファイル１０３を印刷する際には、製本アプリケーション１０４によって電子原稿デスプーラ１０５が起動される。この電子原稿デスプーラ１０５は、指定されたブックファイルをハードディスクから読み出し、そのブックファイルに記述された形式で各ページを印刷するために、前述したOSの出力モジュールに適合する出力コマンドを生成し、不図示の出力モジュールに出力する。その際に、出力デバイスとして使用されるプリンタ１０７用のプリンタドライバ１０６がデバイスドライバとして指定される。そして出力モジュールは、指定されたプリンタ１０７のプリンタドライバ１０６を用いて、受信した出力コマンドをプリンタ１０７で解釈実行可能なデバイスコマンドに変換する。そして、このデバイスコマンドは、プリンタ１０７に送信され、プリンタ１０７によって、そのコマンドに応じた画像が印刷される。

【００１８】

図２は、図１に示すデジタルコンピュータ１００のハードウェア構成を示すブロック図である。

【００１９】

図２において、CPU ２０１は、ROM ２０３のプログラム用ROMに記憶された、或い

はハードディスク211からRAM202にロードされたOSや一般アプリケーション101、製本アプリケーション104などのプログラムを実行し、図1のソフトウェア構成や、後述するフローチャートの手順を実現する。RAM202はCPU201の主メモリ、ワークエリア等として機能する。キーボードコントローラ(KBC)205は、キーボード209や不図示のポインティングデバイスからのキー入力を制御する。CRTコントローラ(CRTC)206は表示部210の表示を制御する。尚、この表示部210はCRTに限定されず、例えば液晶、プラズマなどでも良い。ディスクコントローラ(DKC)207は、ブートプログラム、種々のアプリケーション、フォントデータ、ユーザファイル、後述する編集ファイル等を記憶するハードディスク(HD)211や不図示のフロッピー(登録商標)ディスク(FD)等との間でのアクセスを制御する。PRTC208は、接続されたプリンタ107との間の信号の交換を制御する。NC212はネットワークに接続されて、ネットワークに接続された他の機器との通信制御処理を実行する。

【0020】

＜電子原稿データの形式＞

編集アプリケーション104の詳細に言及する前に、上述した「ブックファイル」のデータ形式を説明する。

【0021】

このブックファイルは、紙媒体の書物を模した3層の層構造を有する。上位層は「ブック」と呼ばれ、1冊の本を模しており、その本全般に係る属性が定義されている。その下の中間層は、本でいう章に相当し、やはり「章」と呼ばれる。各「章」についても、章ごとの属性が定義できる。下位層は「ページ」であり、アプリケーションプログラムで定義された各ページに相当する。各「ページ」についてもページごとの属性が定義できる。尚、一つの「ブック」は複数の「章」を含んでいてよく、また一つの「章」は複数の「ページ」を含むことができる。

【0022】

図3(A)は、ブックファイルの形式の一例を模式的に示す図である。この例のブックファイルにおける、「ブック」、「章」、「ページ」は、それぞれに相当するノードにより示されている。一つのブックファイルは一つの「ブック」を含む。「ブック」、「章」は、「ブック」としての構造を定義するための概念であるから、この定義された属性値と下位層へのリンクとをその実体として含む。「ページ」は、アプリケーションプログラムによって出力された「ページ」ごとのデータを実体として有する。そのため「ページ」は、その属性値の他、原稿ページの实体(原稿ページデータ)と各原稿ページデータへのリンクを含む。尚、紙媒体等に出力する際の印刷ページは、複数の原稿ページを含む場合がある。この構造に関してはリンクによって表示されず、「ブック」、「章」、「ページ」の各階層における属性として表示される。

【0023】

図3(A)において、ブック301には、ブック属性が定義されるとともに、2つの章302A、302Bがリンクされている。このリンクにより、章302A、302Bがブック301に包含されていることが表示される。更に章302Aには、ページ303A、303Bがリンクされ、これらページが含まれることが示されている。各ページ303A、303Bには、それぞれそのページの属性値が定義され、その実体である原稿ページデータ(1)、(2)へのリンクが含まれる。これらリンクは、図3(B)に示す原稿ページデータ304の原稿データ(1)、(2)を指示しており、ページ303A、303Bの実体が、原稿ページデータ(1)、(2)であることを表示する。

【0024】

同様に、章302Bにはページ303C、303Dが含まれ、その実体である原稿ページデータ(3)、(4)へのリンクが含まれている。これらリンクは、図3(B)に示す原稿ページデータ304の原稿データ(3)、(4)を指示しており、ページ303C、303Dの実体が、原稿ページデータ(3)、(4)であることが表示される。

【0025】

図4は、本実施の形態に係るブック属性301を説明するための図である。

【0026】

ここでは、下位層と重複して定義可能な項目に関しては、下位層の属性値が優先的に採用される。そのためブック属性301にのみ含まれる項目に関しては、そのブック属性301で定義された値が、そのブック全体を通して有効な値となる。しかし、下位層と重複する項目については、下位層において定義されていない場合における既定値としての意味を有する。尚、図示された各項目は具体的に1項目に対応するのではなく、関連する複数の項目を含むものもある。

【0027】

図5は、本実施の形態に係る章属性を説明する図、図6は本実施の形態に係るページ属性を説明する図である。ここで、これら章属性とページ属性との関係も、前述のブック属性と下位層の属性との関係と同様である。

【0028】

図4において、ブック属性301に固有の項目は、「印刷方法」、「製本詳細」、「表紙／裏表紙」、「インデックス紙」、「合紙」、「章区切り」の6項目である。これら項目は、そのブックを通して定義される項目である。

【0029】

「印刷方法」の属性としては、「片面印刷」、「両面印刷」、「製本印刷」の3つの値を指定できる。ここで「製本印刷」とは、別途指定する枚数の用紙を束にして2つ折りにし、その束を綴り合わせることで製本が可能となる形式で印刷する方法である。「製本詳細」の属性としては、「印刷方法」で「製本印刷」が指定されている場合に、「見開き方向」や「束になる枚数」等が指定できる。

【0030】

「表紙／裏表紙」の属性は、ブックとしてまとめられる電子原稿ファイルを印刷する際に、表紙および裏表紙となる用紙を付加することの指定、及び付加した用紙への印刷内容の指定を含む。「インデックス紙」の属性は、章の区切りとして、印刷装置に別途用意される耳付きのインデックス紙の挿入の指定およびインデックス（耳）部分への印刷内容の指定を含む。この属性は、印刷用紙とは別に用意された用紙を所望の位置に挿入するインサート機能を持ったインサータが、使用する印刷装置に備えられている場合か、或いは、複数の給紙カセットを使用可能である場合に有効となる。これは「合紙」の属性についても同様である。この「合紙」の属性は、章の区切りとして、インサータから、或いは給紙カセットから供給される用紙の挿入の指定、及び、合紙を挿入する場合には、給紙元の指定などを含む。

【0031】

「章区切り」の属性は、章の区切り目において、新たな用紙を使用するか、新たな印刷ページを使用するか、特に何もしないか等の指定を含む。「片面印刷」時には、新たな用紙の使用と新たな印刷ページの使用とは同じ意味を持つ。「両面印刷」時には、「新たな用紙の使用」を指定すれば連続する章が1枚の用紙に印刷されることは無いが、「新たな印刷ページの使用」を指定すれば、連続する章が1枚の用紙の表裏に印刷されることがあり得る。

【0032】

次に図5に示す章属性に関しては、章に固有の項目はなく、全てブック属性と重複する。従って、章属性における定義とブック属性における定義とが異なれば、章属性で定義された値が優先する。前述のブック属性と章属性とにのみ共通する項目は、「用紙サイズ」、「用紙方向」、「N-up印刷指定」、「拡大縮小」、「排紙方法」の5項目である。このうち「N-up印刷指定」の属性は、1枚の印刷ページに含まれる原稿ページ数を指定するための項目である。ここで指定可能な配置としては、1×1や1×2、2×2、3×3、4×4などがある。「排紙方法」の属性は、排出した用紙にステイプル処理を施すか否かを指定するための項目であり、この属性の有効性は使用する印刷装置がステイプル機能を有するか否かに依存する。

【0033】

次に図6に示すページ属性において、このページ属性に固有の項目には、「ページ回転指定」、「ズーム」、「配置指定」、「アノテーション」、「ページ分割」などがある。「ページ回転指定」は、原稿ページを印刷ページに配置する際の回転角度を指定するための項目である。「ズーム」の属性は、原稿ページの変倍率を指定するための項目である。この変倍率は、仮想論理ページ領域のサイズを100%として指定される。この仮想論理ページ領域とは、原稿ページをN-up等の指定に応じて配置した場合に、1原稿ページが占める領域である。例えば1×1であれば、仮想論理ページ領域は1印刷ページに相当する領域となり、1×2であれば、1印刷ページの各辺を約70パーセントに縮小した領域となる。

【0034】

以上説明した「ブック」、「章」、「ページ」のそれぞれについて共通な属性として、「ウォーターマーク」属性および「ヘッダ・フッタ」属性がある。「ウォーターマーク」とは、アプリケーションで作成されたデータに重ねて印刷される、別途指定される画像や文字列などである。「ヘッダ・フッタ」は、それぞれ各ページの上余白および下余白に印刷されるウォーターマークである。但し、この「ヘッダ・フッタ」には、ページ番号や日時など、変数により指定可能な項目が用意されている。尚、「ウォーターマーク」の属性及び「ヘッダ・フッタ」の属性において指定可能な内容は、「章」と「ページ」では共通であるが、「ブック」の場合では、それらと異なっている。「ブック」においては、「ウォーターマーク」や「ヘッダ・フッタ」の内容を設定できるし、また、ブック全体を通してどのように「ウォーターマーク」や「ヘッダ・フッタ」を印刷するかを指定することができる。一方、「章」や「ページ」では、その章やページにおいて、ブックで設定された「ウォーターマーク」や「ヘッダ・フッタ」を印刷するか否かを指定できるだけである。

【0035】

＜ブックファイルの生成手順＞

上述した製本アプリケーション104によって構造を与えられた電子原稿ファイルである「ブックファイル」は、上述したような構造及び内容を有している。次に、製本アプリケーション104及び電子原稿ライタ102によってブックファイルを作成する手順を説明する。このブックファイルの作成は、製本アプリケーション104によるブックファイルの編集操作の一環として実現される。

【0036】

図7は、本実施の形態に係る製本アプリケーション104によりブックファイルを開く際の手順を説明するフローチャートである。

【0037】

まずステップS701で、開こうとするブックファイルが、新規作成すべきものであるか、それとも既存のものであるかを判定する。新規作成の場合にはステップS702に進み、章を含まないブックファイルを新規に作成する。この新規に作成されるブックファイルは、図3の例で示せば、ブックノード301のみ有し、章のノードに対するリンクが存在しないブックのノードとなる。この場合のブック属性は、新規作成用として予め用意された属性のセットが適用される。そしてステップS704に進み、新規ブックファイルを編集するためのユーザインターフェース（UI）画面を表示する。

【0038】

図11は、新規にブックファイルが作成された際のUI画面の一例を示す図である。この場合には、ブックファイルは実質的な内容を持たないために、UI画面1100には何も表示されない。

【0039】

一方、ステップS701で、既存のブックファイルがあればステップS703に進み、指定されたブックファイルを開き、そのブックファイルの構造、属性、内容に従ってユーザインターフェース（UI）画面を表示する。

【0040】

図10は、この表示されたUI画面の一例を示す図である。このUI画面1100は、ブックの構造を示すツリー部1101と、印刷された状態を表示するプレビュー部1102を含む。ツリー部1101には、そのブックに含まれる章、各章に含まれるページが、前述の図3(A)のような木構造で表示される。ツリー部1101にはページ番号が表示されており、このページ番号は原稿ページの番号を示している。またプレビュー部1102には、印刷ページの内容が縮小されて表示される。その表示順序は、ブックの構造を反映したものとなっている。

【0041】

さて、このオープンされたブックファイルには、電子原稿ライタ102によって、電子原稿ファイルに変換されたアプリケーションデータを、新たな章として追加することができる。この機能を「電子原稿インポート機能」と呼ぶ。前述の図7のフローチャートで示す手順に沿って新規に作成されたブックファイルに、この電子原稿インポートすることで、そのブックファイルに実体が与えられる。この機能は、図10の画面において、アプリケーションデータをドラッグアンドドロップ操作することで起動される。

【0042】

図8は、本実施の形態に係る電子原稿インポートの手順を示すフローチャートである。

【0043】

まず、指定されたアプリケーションデータを生成したアプリケーションプログラムを起動し、デバイスドライバとして電子原稿ライタ102を指定してアプリケーションデータを印刷させることにより電子原稿データに変換する(ステップS801)。この電子原稿データへの変換を終えるとステップS802に進み、その変換されたデータが画像データであるか否かを判定する。この判定は、ウィンドウズ(登録商標)OSの下であれば、アプリケーションデータのファイル拡張子に基づいて行うことができる。例えば、拡張子が「bmp」であればウィンドウズ(登録商標)ビットマップデータであり、「jpg」であればJPEG圧縮された画像データ、「tiff」であればtiff形式の画像データであると判定できる。また、このような画像データの場合はステップS801のようにアプリケーションを起動せずに、画像データから直接電子原稿ファイルを生成することが可能であるため、ステップS801の処理を省略することも可能である。

【0044】

ステップS802で、画像データでなかった場合はステップS803に進み、ステップS801で生成された電子原稿ファイルを、現在開かれているブックファイルのブックに、新たな章として追加する。この場合、章属性としては、ブック属性と共通するものについてはブック属性の値がコピーされ、そうでないものについては、予め用意された規定値に設定される。

【0045】

又ステップS802で画像データである場合にはステップS804に進み、原則として新たな章は追加されず、指定されている章に、ステップS801で生成された電子原稿ファイルに含まれる各原稿ページを追加する。但し、ブックファイルが新規作成されたファイルであれば、新たな章が作成されて、その章に属するページとして、電子原稿ファイルの各ページが追加される。ここでページ属性は、上位層の属性と共通のものについてはその属性値が与えられ、アプリケーションデータにおいて定義された属性を電子原稿ファイルに引き継いでいるものについてはその値が与えられる。例えば、「N-up印刷指定」などがアプリケーションデータにおいてされていた場合には、その属性値が引き継がれる。このようにして、新規なブックファイルが作成され、或いは新規な章が追加される。

【0046】

図9は、図8のステップS801において、電子原稿ライタ102により電子原稿ファイルを生成させる手順を示すフローチャートである。

【0047】

まずステップS901で、新たな電子原稿ファイルを作成してそれを開く。次にステップS902に進み、その指定したアプリケーションデータに対応するアプリケーションを起

動し、電子原稿ライタ102をデバイスドライバとして、OSの出力モジュールに対して出力コマンドを送信させる。この出力モジュールは、受信した出力コマンドを電子原稿ライタによって電子原稿形式のデータに変換して出力する。ここで、その出力先は、ステップS901で開いた電子原稿ファイルである。次にステップS903に進み、指定されたデータの全てについて変換が終了したか判定し、終了していればステップS904に進み、その電子原稿ファイルを閉じる。この様にして電子原稿ライタ102によって生成される電子原稿ファイルは、前述の図3(B)に示した、原稿ページデータの実体を含むファイルである。

【0048】

＜ブックファイルの編集＞

以上のようにして、アプリケーションデータからブックファイルを作成することができる。こうして生成されたブックファイルについては、章及びページに対して次のような編集操作が可能である。

- (1) 新規追加
- (2) 削除
- (3) コピー
- (4) 切り取り
- (5) 貼り付け
- (6) 移動
- (7) 章名称変更
- (8) ページ番号名称振り直し
- (9) 表紙挿入
- (10) 合紙挿入
- (11) インデックス紙挿入
- (12) 各原稿ページに対するページレイアウト。

【0049】

この他、一旦行った編集操作を取り消す操作や、更に、取り消した操作をやり直す操作が可能である。これら編集機能により、例えば複数のブックファイルの統合、ブックファイル内で章やページの再配置、ブックファイル内で章やページの削除、原稿ページのレイアウト変更、合紙やインデックス紙の挿入などといった編集操作が可能となる。これらの操作を行うと、図4乃至図6に示す属性に操作結果が反映されたり、或いはブックファイルの構造に反映される。例えば、ブランクページの新規追加操作を行えば、指定された箇所にブランクページが挿入される。このブランクページは原稿ページとして扱われる。また、原稿ページに対するレイアウトを変更すれば、その変更内容は、印刷方法やN-up印刷、表紙／裏表紙、インデックス紙、合紙、章区切りといった属性に反映される。

【0050】

＜ブックファイルの出力＞

以上のように作成・編集されるブックファイルは印刷出力を最終目的としている。利用者が図10に示す製本アプリケーションのUI画面1100からファイルメニューを選択し、そこから印刷を選択すると、指定した出力デバイスにより印刷が行われる。この際、まず製本アプリケーション104は、現在開かれているブックファイルからジョブチケットを作成して電子原稿デスプーラ105に渡す。これにより電子原稿デスプーラ105は、ジョブチケットをOSの出力コマンド、例えばウィンドウズ（登録商標）のGDIコマンドに変換し、それを出力モジュール、例えばGDIに送信する。出力モジュールは、その指定されたプリンタドライバ106によって、出力デバイスに適したコマンドを生成し、その出力デバイスに送信する。

【0051】

ここでジョブチケットは、原稿ページを最小単位とする構造を有するデータである。このジョブチケットにおけるデータ構造は、用紙上における原稿ページのレイアウトを定義している。このジョブチケットは、1ジョブにつき1つ発行される。そのため、まず最上位

にドキュメントというノードがあり、文書全体の属性、例えば両面印刷／片面印刷などが定義されている。その下には、用紙ノードが属し、用いるべき用紙の識別子や、プリンタにおける給紙口の指定などの属性が含まれる。各用紙ノードには、その用紙で印刷されるシートのノードが属する。1シートは1枚の印刷用紙に相当し、各シートには、印刷ページ（物理ページ）が属している。片面印刷ならば1シートには1物理ページが属し、両面印刷ならば1シートに2物理ページが属する。各物理ページには、その上に配置される原稿ページが属する。また物理ページの属性として、原稿ページのレイアウトが含まれる。

【0052】

電子原稿デスプーラ105は、上述のジョブチケットを出力モジュールへの出力コマンドに変換する。

【0053】

＜その他のシステム構成＞

本実施の形態に係る文書処理システムの概要は以上のようなものである。これはスタンドアロン型のシステムであるが、これを拡張したサーバ・クライアントシステムでもほぼ同様の構成・手順でブックファイルが作成・編集される。ただし、ブックファイルや印刷処理はサーバによって管理される。

【0054】

図12は、本実施の形態に係るサーバ・クライアント型の文書処理システムの構成を示すブロック図である。

【0055】

クライアント文書処理システム1200は、スタンドアロン型システムに、クライアントモジュールであるDOMS (Document Output Management Service: 文書出力管理サービス) ドライバ109およびDOMSプリントサービスモジュール110、DS (文書サービス) クライアントモジュール108を加えた構成を有する。このクライアント文書処理システム1200に、文書管理サーバ1201および印刷集中管理サーバ1202およびプリントサーバ1203が接続されている。これらサーバは、通常ネットワークによってクライアント文書処理システム1200と接続されるが、サーバが同時にクライアントとしても機能する場合には、ネットワーク間の通信をシミュレートするプロセス間通信によって接続される。尚、図12では、文書管理サーバ1201と印刷集中管理サーバ1202の両サーバがクライアントに接続されているが、いずれか一方のみがネットワーク上に存在する場合もあり得る。接続されているサーバが文書管理サーバであれば、そのクライアントモジュールを含む文書管理サーバクライアントシステム1201SCが、印刷集中管理サーバ1202であれば、そのクライアントモジュールを含む印刷管理サーバクライアントシステム1202SCが、スタンドアロン型文書管理システムに追加される。

【0056】

文書管理サーバ1201は、製本アプリケーション104により作成・編集されたブックファイルを格納するサーバである。この文書管理サーバ1201によってブックファイルを管理する場合、ブックファイルは、クライアントPCのローカルHDに代わって、或いはそれに加えて、文書管理サーバ1201のデータベース1211に保存される。製本アプリケーション104と文書管理サーバ1201との間のブックファイルの保存及び読み出しは、DSクライアント108及びDSコア1212を介して行われる。

【0057】

印刷集中管理サーバ1202は、クライアント文書管理システム1200に格納された、或いは文書管理サーバ1201に格納されたブックファイルの印刷を管理するサーバである。クライアントにおける印刷要求は、DOMSドライバ109及びDOMSプリントサービスモジュール110を介して印刷集中管理サーバ1202のDOMSWGサーバモジュール1221に送信される。印刷集中管理サーバ1202は、クライアントのプリンタ107で印刷する場合には、クライアントのDOMSプリントサービスモジュール110を介して電子原稿デスプーラ105に電子原稿データを渡す。

【0058】

一方、プリントサーバ1203により印刷する場合には、プリントサーバ1203のDOWSプリントサービスモジュール1231に送信する。印刷集中管理サーバ1202は、例えば保存されているブックファイルに対して、その印刷要求を発行した利用者の資格などについてセキュリティチェックを行ったり、印刷処理のログを保存したりする。このように、この文書処理システムは、スタンドアロンとしても、クライアントサーバシステムとしても実現できる。

【0059】

＜プレビュー表示の内容＞

既に説明したとおり、ブックファイルが製本アプリケーションによって開かれると、図10に示すユーザインターフェース画面1100が表示される。ここでツリー部1101には、開いているブック（以下、「注目ブック」と呼ぶ）の構造を示すツリーが表示される。印刷プレビュー部1102には、利用者の指定に応じて、3通りの表示方法が用意されている。第1は原稿ページをそのまま表示する「原稿ビューモード」である。この「原稿ビューモード」では、注目ブックに属する原稿ページの内容が縮小されて表示される。この場合、プレビュー部1102の表示には、その原稿のレイアウトは反映されない。第2は「印刷ビューモード」である。この「印刷ビューモード」では、プレビュー部1102には、各原稿ページのレイアウトが反映された形で原稿ページが表示される。第3は「簡易印刷ビューモード」である。この「簡易印刷ビューモード」では、各原稿ページの内容はプレビュー部1102の表示には反映されず、そのレイアウトのみが反映される。

【0060】

図10の画面においては、メニュー欄から「印刷形式」を選択すると、印刷設定を変更することができる。サブメニューとして、さらにページについてや章についてあるいはブックの、いずれについての設定を変更するかオペレータは選択できる。そして、いずれかが選択されると、選択された階層の属性、たとえばブックが選択されればブック属性に含まれる印刷設定項目を変更するための画面が表示される。表示される画面は、選択された属性の階層に応じて、図4乃至図6の各項目を設定させる画面である。その中において、オペレータは所望の印刷設定を入力することができる。オペレータに入力された印刷設定は、指定された階層に応じて、図4乃至図6に示す属性の一部として格納される。

【0061】

また、複数の階層において重複して設定可能な項目については、例えば、下位の属性において「例外指定」である旨をまず指定し、その上で値を指定する。たとえば、図5の章属性に含まれる用紙方向の項目は、ブック属性にも重複して含まれている。したがって、章属性において「用紙方向」をブックの例外として指定する場合には、図10のメニューから「印刷形式」を選択し、さらにサブメニューから「章の指定」を選択する。オペレータはさらに、そのときに表示されるユーザインターフェース画面から「用紙方向」について「例外指定」をまず指定し、その後で所望の用紙方向を章について設定する。

【0062】

このようにして、文書編集集中においても、印刷設定をブック、章、ページごとに、オペレータが所望の値に設定することができる。なお、この印刷設定の手順は一例であり、結局のところ、図4乃至図6に設定値が反映され、さらにそれが印刷結果に反映されるものであれば、その手順そのものは本発明に関する限りどのようなものでもよい。

【0063】

次に本発明の実施の形態の特徴について説明する前に、本実施の形態に適用可能なホストコンピュータ100（本発明の情報処理装置に相当する）とプリンタ107とを有する文書処理システムの構成について説明する。尚、ホストコンピュータ100（以下、クライアントと呼ぶこともある）の構成に関しては、前述の図2を参照して説明した通りであるので、ここでは共通する箇所には同じ番号を付し、その説明を省略する。またプリンタ107は、クライアントに接続されたローカルプリンタであっても、或いはサーバに接続されたプリンタ1204（図12）の場合であってもいずれの場合にも適用可能であるもの

とする。

【0064】

＜本実施形態の文書処理システム＞

図13は、本発明の実施の形態に係る文書処理システムの構成を説明するためのブロック図である。なお、本実施の形態に係る機能が実行されるのであれば、単体の機器であっても、複数の機器からなるシステムであっても、LAN、WAN等のネットワークを介して接続がなされ処理が行われるシステムであっても本発明を適用できる。尚、ここでホストコンピュータ100の構成は前述の図2と同じであるため、その説明を省略する。

【0065】

プリンタ107はCPU12により制御される。このCPU12は、ROM13のプログラム用ROMに記憶された制御プログラム、或いは外部メモリ14に記憶されRAM19にロードされた制御プログラムに基づいて、システムバス15に接続される印刷部（プリンタエンジン）17に出力情報としての画像信号を出力する。また、このROM13のプログラム用ROMには、CPU12の制御プログラムが記憶されている。ROM13のフォント用ROMには、上記出力情報を生成する際に使用するフォントデータ等が記憶され、ROM13のデータ用ROMには、ハードディスク等の外部メモリ14が設けられていないプリンタの場合には、ホストコンピュータ100で利用される情報等が記憶されている。

【0066】

CPU12は入力部18を介してホストコンピュータ100との間での通信処理が可能となっており、プリンタ107の情報等をホストコンピュータ100に通知できる。RAM19は、CPU12の主メモリや、ワークエリア等として機能するRAMで、図示しない増設ポートに接続されるオプションRAMにより、そのメモリ容量を拡張することができるように構成されている。なお、RAM19は、出力情報の展開領域、環境データ格納領域、NVRAM等に用いられる。前述したハードディスク（HD）、ICカード等の外部メモリ14は、メモリコントローラ（MC）20により、そのアクセスが制御されている。外部メモリ14はオプションとして接続され、フォントデータ、エミュレーションプログラム、フォームデータ等を記憶する。また操作パネル21には、操作のためのスイッチおよびLED表示器等が配されている。

【0067】

また、前述した外部メモリ14は1個に限らず、複数個備えられ、内蔵フォントに加えてオプションカード、言語系の異なるプリンタ制御言語を解釈するプログラムを格納した外部メモリを複数接続できるように構成されていてもよい。更に、図示しないNVRAMを有し、操作パネル21からのプリンタモード設定情報を記憶するようにしてもよい。

【0068】

ここで、ホストコンピュータ100は、図1を参照して前述したファイル、アプリケーション、デスプーラ及びドライバ等を備えているものとする。そして、このアプリケーション101により生成された文書ファイル（アプリケーションデータ）を電子原稿ライタ102を介することにより電子原稿ファイル103が生成される。電子原稿ファイル103は前述したように、製本アプリケーション104によりファイルオープンすることが可能であり、製本アプリケーション104は、例えば前述の図10に示すような、操作画面を表示部210に表示させる。製本アプリケーション104と汎用の一般アプリケーション101との大きな違いは、製本アプリケーション104が、通常プリンタドライバ106が提供するはずの印刷設定機能を有していることである。この印刷設定機能により、製本アプリケーション104は、汎用アプリケーション101とは異なり、文書のページ順の入れ替えや複製（コピー）、削除などの編集機能に加え、ステイプル・パンチ穴などの、プリンタ107における機能設定を電子原稿ファイルに対して行うことができ、指定したプリンタを使用して印刷することが可能である。なお、本実施の形態で説明する製本アプリケーション104を、印刷設定アプリケーションとも呼ぶ。

【0069】

<テンプレート機能>

ここで、従来製本アプリケーション104で提供されているテンプレート機能について説明する。上述したように製本アプリケーション104では、電子原稿ライタ102を介することにより生成された電子原稿ファイルに対して、出力用紙、ページレイアウト、製本などの設定編集を行うことにより、図4で示した属性をもった文書ファイルを作成することができるが、さらにこのような設定情報をテンプレートとして登録し、文書の新規作成時には、あらかじめ用意されたテンプレートの中から好きなテンプレートを選択することができる機能を提供している。

【0070】

このテンプレートのデータはクライアントPCのレジストリに登録して管理しており、製本アプリケーション104は、ユーザが任意にテンプレートデータを変更できるような編集機能も提供している。レジストリとは、各種アプリケーション等の環境設定やドライバの指定などの情報を保存しているファイルである。ウインドウズ（登録商標）オペレーティングシステムでは、バイナリ・ファイルとして保存されているために、汎用のテキストエディタ等では変更することが不可能となっている。製本アプリケーション104は、このバイナリファイルを読んで、ユーザに編集可能なデータとして提示し、編集を行わせる機能を有している。

【0071】

さらに、製本アプリケーション104は、テンプレート情報をファイルにして保存する機能を持ち、テンプレートファイルを他のPCへ持ち出す、または他のPCからテンプレートデータを取り込むことを可能とする、インポート／エクスポート機能も提供されている。

【0072】

図14は新規に文書ファイルを生成する際に、製本アプリケーション104において表示されるテンプレート選択のためのウィンドウの表示例である。ユーザはこの画面で製本アプリケーション104が用意しているテンプレートの中から好きなテンプレートを選択することが可能である。

【0073】

次に、図15は製本アプリケーション104において文書ファイルを編集中に、現在のテンプレートを変更するための「テンプレート選択」ウィンドウ1501の表示例である。ユーザは、このウィンドウで編集中にテンプレートを変更することも可能である。さらに、ユーザは、このテンプレートを新たに作成し登録することも可能である。

【0074】

<従来のテンプレート設定操作>

図16は、製本アプリケーション104において、テンプレートを編集するための「テンプレート設定」ウィンドウ表示の一例である。一覧表示欄1605には、製本アプリケーション104で用意しているテンプレートの一覧が表示される。ここで、オペレータがテンプレートを選択し、「編集」ボタン1601をクリックすると、図17のテンプレート編集画面が表示され、選択したテンプレートの設定を自由に変更することが可能である。また、「追加」ボタン1602をクリックした場合も、図17の「テンプレート編集」画面が表示され、ここで表示されているテンプレートの設定はデフォルト初期値（1UP、片面、フィニッシングなし）になっているので、ここでユーザは、新しいテンプレート名称を指定し、個々の設定内容を変えて新しいテンプレートを作成し、クライアントPCに登録することができる。

【0075】

なお、図16のユーザインターフェース画面においてエクスポートボタン1604を選択すると、クライアントPCのレジストリに登録されているテンプレートの情報をファイルとして保存することができる。また、インポートボタン1603を選択すれば、テンプレートファイルを新たにクライアントPCのレジストリに登録することができる。

【0076】

しかし、この従来のシステムでは、ユーザがテンプレートを新しく作成する場合には、アプリケーションで始めから登録されているテンプレートを自分の好みの属性値に変えるか、または新しくデフォルト値（例えば、1UP、片面、フィニッシングなしのような設定）を初期値として作成するという方法しか提供されていなかった。

【0077】

＜編集中の文書の設定値を利用したテンプレートの作成＞

続いて、本発明にかかる、現在編集中の文書ファイルの設定値を利用したテンプレートの作成機能について説明する。

【0078】

図18は、製本アプリケーション104において、文書ファイルを編集中に「テンプレートの設定」ウィンドウを表示した例である。「新規追加」ボタン1802は、図16の「追加」ボタン1602と同じ機能を持つ。また「削除」ボタン1802、「編集」ボタン1803、「インポート」ボタン1804、「エクスポート」ボタン1805も、従来と同じ機能である。しかし、従来とは異なり、新たに「現在の設定を登録」ボタン1801が追加されている。ここをクリックすると、図19の「現在の設定を登録」ウィンドウ1902が表示される。このとき「現在の設定を登録」ウィンドウ1902で示す「設定内容」の表示欄1903には、現在編集中の文書ファイルで設定されている内容が表示されており、ユーザはテンプレート名称1901を指定するだけで、現在編集中の文書ファイルの設定をもとにテンプレートを作成することが可能である。

【0079】

以上のユーザインターフェース画面及び操作手順により現在編集中の文書ファイルの設定をテンプレートとして保存し、またテンプレートを用いて新規文書を作成することができる。次に、上述した操作に伴って印刷制御アプリケーション104により実行される処理手順と、テンプレートの内容について説明する。

【0080】

本実施形態においては、テンプレートは図4に示すブック属性そのものである。すなわち、テンプレートに含まれる設定項目は図4に示すブック属性に含まれる項目と一致する。上記操作により、電子原稿ファイル103から、図3に示す構造を有する編集中の文書ファイルのブック属性が読み出されて、必要に応じてレジストリに格納する形式に変換され、それがレジストリに登録される。図20はテンプレートの設定メニューをオペレータが選択した際の、製本アプリケーション104によるテンプレート設定処理手順のフローチャートである。

【0081】

まず、登録されたテンプレートについて一定の順序で注目し、注目テンプレートの内容を読む（S2101）。そして、テンプレートの内容、すなわち印刷設定に応じたアイコンと注目テンプレート名とをテンプレート一覧表示欄に表示する（S2102）。このテンプレート一覧表示欄が、図18における「テンプレート一覧」のウィンドウに相当する。そして表示される各アイコン及びテンプレート名は、テンプレート一覧に表示された各アイコンに相当する。ここでは、たとえば、「両面印刷／片面印刷」や、「インデックス紙」の有無、「用紙方向」等の設定に応じてその設定を直感的に示すアイコンが選択されて表示される。以上のステップを未表示のテンプレートがなくなるまで次々と異なるテンプレートに注目して繰り返す（S2103）。これらアイコンとテンプレートデータとはリンクされているため、一のアイコンが選択されれば一意的に対応するテンプレートも決定される。

【0082】

「テンプレート一覧」の表示を終了したなら、その時点で選択されているテンプレートの設定内容を、内容表示欄に表示する（S2104）。内容表示欄は、図18の「テンプレート一覧」の右側にある表示欄であり、選択されたテンプレートに含まれる各設定項目（図4に従っていえば属性の項目）について、その値が具体的に表示される。図18の例では、選択されている「A4縦（1）」なる名称のテンプレートにおいては、用紙サイズは

A4サイズ、用紙方向は縦、ページレイアウト（図4においてはN-u p印刷）は1ページ／枚などと設定されており、それらの設定値が表示されている。

【0083】

ステップS2104は、図18の画面が表示された後、オペレータがテンプレート一覧の中から所望のアイコンを選択した場合にも新たに選択されたアイコンについて繰返し実行され、対応するテンプレートの内容が表示される。

【0084】

図18の画面において「現在の設定を登録」ボタン1802が押されると、図21の処理が製本アプリケーションにより実行される。

【0085】

まず、現在編集中の文書（ブック）ファイルに設定された「ブック属性」（図4参照）を、電子原稿ファイル103から読み出す（S2201）。そしてその設定内容を、図19に示した「現在の設定を登録」ウインドウ1902の「設定内容」の表示欄1903に表示する（S2202）。

【0086】

この後オペレータによる入力待ちとなり（S2203）、入力があればその内容を判定する（S2204）。テンプレート名1901が入力されたら次の入力を待ち、「OK」ボタンならば入力されているテンプレート名を付して、ステップS2201で読み込んだブック属性をテンプレートとしてレジストリに登録する（S2205）。レジストリには、階層的に構成されるレジストリキーと、レジストリキーに対応するエントリ値とが含まれている。そこで、たとえば、製本アプリケーションに相当するレジストリキーに属するサブキーとしてテンプレートを指定された名称で登録し、そこに含まれるエントリ値として、各設定項目名と対応する値とを登録する。

【0087】

またもしステップS2205の時点で未だテンプレート名が入力されていなければ、あらためてその入力を要求する。一方「キャンセル」ボタンが押された場合には、読み込んだブック属性を破棄して処理を終了する。

【0088】

なお、図21においては説明の便宜上一連の処理の中で入力待ちをしているが、入力待ちの際にはいったんタスクを終了する方法もある。

【0089】

以上のようにして、編集中の文書ファイルのブック属性を新規のテンプレートとして登録することができる。このようにして登録したテンプレートは、図15に示す「テンプレートの選択」メニューを指定することで、「テンプレート選択」ウインドウ1501が表示され、その中にテンプレートのアイコンが表示される。それをオペレータが選択することで、新規の文書ファイルが作成され、そのブック属性として選択されたテンプレートの設定項目の値が複写される。

【0090】

なお新規文書ファイルの作成時に限らず、既存の文書ファイルのブック属性を、選択したテンプレートの内容で置換することもできる。

【0091】

以上の構成及び処理手順により、本実施形態に係る文書処理システムでは、編集中の文書ファイルの設定情報、特にブック属性に含まれる設定情報をもとにテンプレートを作成することが可能となった。それにより、ユーザは、テンプレート作成のためにあらためて設定編集を行う手間が省け、新しいテンプレートファイルを容易に登録することが可能となった。

【0092】

なお、本実施形態では、製本アプリケーションと呼ばれているプログラムは製本のためのみならず、文書の体裁を利用者の望むように整形する機能を有するアプリケーションであり、その機能に従えば、文書編集アプリケーションと呼ぶこともできる。

【0093】

[変形例1]

上記実施形態においては、ブック属性に含まれる設定項目のみがテンプレートに反映されている。しかしながら、章属性(図5)やページ属性(図6)に含まれる項目の値をテンプレートに反映するようにしてもよい。この場合、一つの方法として、以下の通りにテンプレートを作成できる。

【0094】

図4乃至図6に示すように、各階層の属性に含まれる設定項目には重複する項目がある。重複する項目について、上位階層と下位階層とで異なる設定がされている場合、下位階層における設定は、例外的な値を示している。たとえば、図5に示す章属性に含まれる項目はすべてブック属性と重複している。ブック属性に含まれる項目は、そこに設定された値が文書全体に適用される。それに対して、章属性において設定された項目があれば、その章については、ブック属性において設定された値とは異なる、例外的な設定がされていることを示す。たとえば文書ファイルAについて、そのブック属性に含まれる用紙サイズ項目の値が「A4」であって、その文書ファイルに属する章A1の章属性に含まれる用紙サイズ項目の値が「A3」であれば、文書全体は用紙サイズA4で印刷されるものの、章A1については例外的にA3で印刷されることを示す。このように、重複する項目については、下位階層の項目において値が設定されていれば、その値により上位階層の設定値が無視される結果となる。

【0095】

そこで、現在編集中の文書において、更にその中から特定の章が選択されている場合には、ブック属性に含まれる設定項目の値を、選択されている章の章属性に含まれる、ブック属性と重複する項目の設定値で置換したものを、テンプレートとして保存する。この結果テンプレートに含まれる項目は図4に示すものとなるが、設定される値には選択された章における設定値が反映される。

【0096】

このように構成することで、編集中の文書の内、選択した章の章属性を反映したテンプレートを容易に作成することができる。

【0097】

[変形例2]

上記変形例はページ属性についても適用できる。ページ属性と章属性及びブック属性との関係は、上述した章属性とブック属性との関係と同様である。そこで、現在編集中の文書において、更にその中から特定のページが選択されている場合には、ブック属性に含まれる設定項目の値を、選択されているページの含まれる章の章属性における、ブック属性と重複する項目の設定値で置換して中間データを作成し、さらに、選択されているページのページ属性において重複する項目が設定されていれば、その設定値で前述した中間データを置換して、それをテンプレートとして保存する。この結果テンプレートに含まれる項目は図4に示すものとなるが、設定される値には選択されたページにおける設定値が反映される。

【0098】

このように構成することで、編集中の文書の内、選択したページのページ属性を反映したテンプレートを容易に作成することができる。

【0099】

[変形例3]

なお本発明は、複数の機器(例えばホストコンピュータ、インターフェース機器、リーダー、プリンタなど)から構成されるシステムに適用しても、一つの機器からなる装置(例えば、複写機、ファクシミリ装置など)に適用してもよい。

【0100】

また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体(または記録媒体)を、システムあるいは装置に供給し、そのシ

システムあるいは装置のコンピュータ（またはCPUやMPU）が記憶媒体に格納されたプログラムコードを読み出し実行することによっても達成される。この場合、記憶媒体から読み出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているオペレーティングシステム（OS）などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれる。

【0101】

更に、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張カードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張カードや機能拡張ユニットに備わるCPUなどが実際の処理の一部又は全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれる。

【0102】

〔その他の態様〕

〔実施態様1〕 文書を印刷する際の印刷設定を変更可能な文書処理方法であって、編集中の文書の印刷設定をテンプレート情報として登録する登録工程と、前記登録工程により登録されたテンプレート情報を、編集中の文書または新規な文書の印刷設定として読み込む読み込み工程とを備えることを特徴とする文書処理方法。

【0103】

〔実施態様2〕 前記登録工程により登録されているテンプレートの内容を編集するテンプレート編集工程を更に備えることを特徴とする実施態様1に記載の文書処理方法。

【0104】

〔実施態様3〕 前記文書はページを葉とする木構造を有し、各部分木ごとに印刷設定を設定可能であり、前記登録工程は前記木構造の全体について設定された印刷設定をテンプレートとして登録することを特徴とする実施態様1に記載の文書処理方法。

【0105】

〔実施態様4〕 前記文書はページを葉とする木構造を有し、各部分木ごとに印刷設定を設定可能であるとともに、重複する設定項目についてはより下位の設定が優先適用され、前記登録工程は選択された部分木に対して適用される印刷設定をテンプレートとして登録することを特徴とする実施態様1に記載の文書処理方法。

【0106】

〔実施態様5〕 前記文書を編集する文書編集工程を更に備え、該文書編集工程により印刷設定を変更可能であることを特徴とする実施態様1乃至4のいずれか1項に記載の文書処理方法。

【0107】

〔実施態様6〕 前記印刷設定には、印刷方法、用紙方向、綴じ代及び綴じ方向、ページレイアウト、拡大縮小、ウオータマーク、ヘッダ及びフッタ、排紙方法、製本設定、表紙及び裏表紙、インデックス紙、合紙、章区切りの、少なくともいずれかが含まれることを特徴とする実施態様1乃至5のいずれか1項に記載の文書処理方法。

【0108】

〔実施態様7〕 文書を印刷する際の印刷設定を変更可能な文書処理装置であって、編集中の文書の印刷設定をテンプレート情報として登録する登録手段と、前記登録手段により登録されたテンプレート情報を、編集中の文書または新規な文書の印刷設定として読み込む読み込み手段とを備えることを特徴とする文書処理装置。

【0109】

〔実施態様8〕 前記登録手段により登録されているテンプレートの内容を編集するテン

プレート編集手段を更に備えことを特徴とする実施態様7に記載の文書処理装置。

【0110】

〔実施態様9〕 前記文書はページを葉とする木構造を有し、各部分木ごとに印刷設定を設定可能であり、前記登録手段は前記木構造の全体について設定された印刷設定をテンプレートとして登録することを特徴とする実施態様7に記載の文書処理装置。

【0111】

〔実施態様10〕 前記文書はページを葉とする木構造を有し、各部分木ごとに印刷設定を設定可能であるとともに、重複する設定項目についてはより下位の設定が優先適用され、前記登録手段は選択された部分木に対して適用される印刷設定をテンプレートとして登録することを特徴とする実施態様7に記載の文書処理装置。

【0112】

〔実施態様11〕 前記文書を編集する文書編集手段を更に備え、該文書編集手段により印刷設定を変更可能であることを特徴とする実施態様7乃至10のいずれか1項に記載の文書処理装置。

【0113】

〔実施態様12〕 前記印刷設定には、印刷方法、用紙方向、綴じ代及び綴じ方向、ページレイアウト、拡大縮小、ウオータマーク、ヘッダ及びフッタ、排紙方法、製本設定、表紙及び裏表紙、インデックス紙、合紙、章区切りの、少なくともいずれかが含まれることを特徴とする実施態様7乃至11のいずれか1項に記載の文書処理装置。

【0114】

〔実施態様13〕 コンピュータにより、実施態様1乃至6のいずれか1項に記載の文書処理方法を実行させることを特徴とするプログラム。

【0115】

〔実施態様14〕 実施態様13に記載のプログラムを記録したことを特徴とするコンピュータ可読記録媒体。

【0116】

【発明の効果】

以上説明したように本発明によれば、テンプレートを作成する際に、現在編集中の文書ファイルの設定情報をもとにテンプレートを作成することが可能となった。それにより、ユーザは、テンプレート作成のために設定編集を行う手間が省け、新しいテンプレートファイルを容易に登録することが可能となった。

【図面の簡単な説明】

【図1】 本発明の実施の形態に係るスタンドアロン型の文書処理システムの構成を説明するためのブロック図である。

【図2】 本実施の形態に係る文書処理システムのホストコンピュータの構成を示すブロック図である。

【図3】 本実施の形態に係るブックファイルの構造の一例を示す図で、図3（A）はブックファイルの形式の一例を模式的に示す図、図3（B）は原稿データの構成を示す図である。

【図4】 本実施の形態に係るブック属性の一例を示す図である。

【図5】 本実施の形態に係る章属性の一例を示す図である。

【図6】 本実施の形態に係るページ属性の一例を示す図である。

【図7】 本実施の形態に係るブックファイルを開く手順を説明するフローチャートである。

【図8】 本実施の形態に係る、電子原稿ファイルをブックファイルにインポートする手順を説明するフローチャートである。

【図9】 本実施の形態に係る、アプリケーションデータを電子原稿ファイルに変換する手順を説明するフローチャートである。

【図10】 既存のブックファイルを開いた際のユーザインターフェース画面の一例を示す図である。

【図11】新規のブックファイルを開いた際のユーザインターフェース画面の一例を示す図である。

【図12】本実施の形態に係るクライアントサーバ型の文書処理システムの構成を示すブロック図である。

【図13】本実施の形態に係る文書処理システムにおけるホストコンピュータとプリンタの構成を説明するためのブロック図である。

【図14】本実施の形態に係る文書処理システムにおける、従来のテンプレート選択のためのユーザインターフェース画面の一例を示す図である。

【図15】本実施の形態に係る文書処理システムにおける、従来のテンプレート選択のためのユーザインターフェース画面を示す図である。

【図16】本実施の形態に係る文書処理システムにおける、従来のテンプレート設定のためのユーザインターフェース画面を示す図である。

【図17】本実施の形態に係る文書処理システムにおける、従来のテンプレート編集のためのユーザインターフェース画面の一例を示す図である。

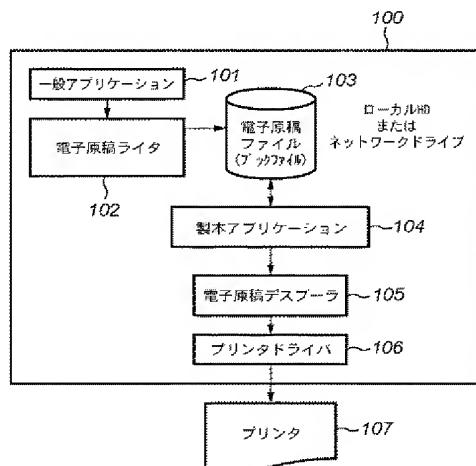
【図18】本発明実施の形態に係る文書処理システムにおける、テンプレート設定のためのユーザインターフェース画面の一例を示す図である。

【図19】実施の形態に係る文書処理システムにおける、テンプレート設定のためのユーザインターフェース画面の一例を示す図である。

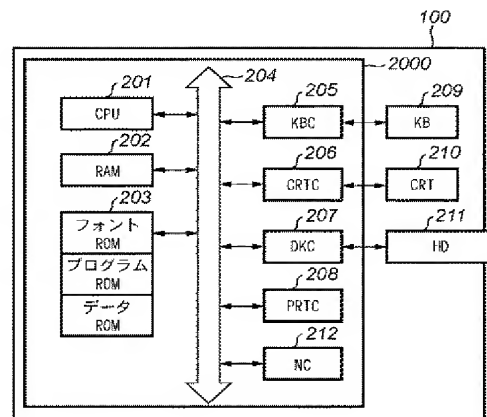
【図20】実施の形態に係る文書処理システムにおける、製本アプリケーションによるテンプレートの設定手順のフローチャートである。

【図21】実施の形態に係る文書処理システムにおける、製本アプリケーションによるテンプレートの設定手順のフローチャートである。

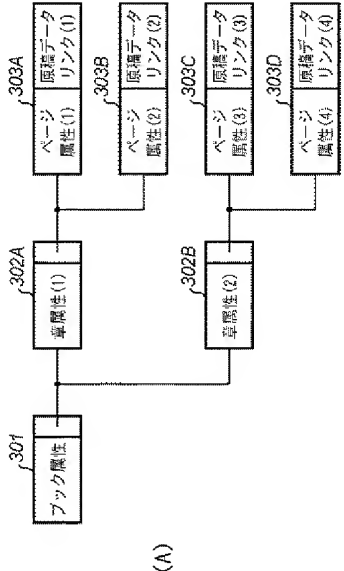
【図1】



【図2】



【図3】



(B)

| | | | |
|-----------|-----------|-----------|-----------|
| 原稿データ (1) | 原稿データ (2) | 原稿データ (3) | 原稿データ (4) |
|-----------|-----------|-----------|-----------|

【図4】

| NO | 属性情報 | 備考 |
|----|----------|---|
| 1 | 印刷方法 | |
| 2 | 用紙サイズ | 用紙サイズ/面紙サイズ |
| 3 | 用紙方向 | 縦/横 |
| 4 | とじ代とし方向 | ページ数/部数/境界線/配置位置等 |
| 5 | Wd印刷 | 0%OFF |
| 6 | 拡大縮小 | 用紙サイズに同じサイズのままは印刷を選択した場合は自動で0%に指定、0%に指定された場合は印刷単位に個別の指定 |
| 7 | ウォーダーマーク | ウォーダーマーク |
| 8 | ヘッダ・フッタ | ヘッダ・フッタ |
| 9 | 排紙方法 | ステイプル/パンチ穴 |
| 10 | 紙本仕様 | 書き方向/中とし/縦横指定/とじ代/分冊指定等 |
| 11 | 表紙・裏表紙 | 表紙/裏表紙 |
| 12 | インデックス紙 | インデックス紙 |
| 13 | 合巻 | 合巻 |
| 14 | 裏面切り | 裏面切り |

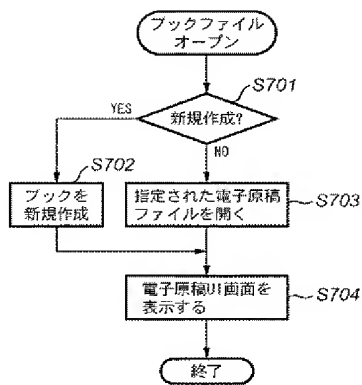
【図5】

| NO | 属性情報 | 備考 |
|----|----------|---|
| 1 | 用紙サイズ | 用紙サイズ/面紙サイズ |
| 2 | 用紙方向 | 縦/横 |
| 3 | Wd印刷指定 | 0%OFF |
| 4 | 拡大縮小 | 用紙サイズに同じサイズのままは印刷を選択した場合は自動で0%に指定、0%に指定された場合は印刷単位に個別の指定 |
| 5 | ウォーダーマーク | ウォーダーマーク |
| 6 | ヘッダ・フッタ | ヘッダ・フッタ |
| 7 | 排紙方法 | ステイプル |

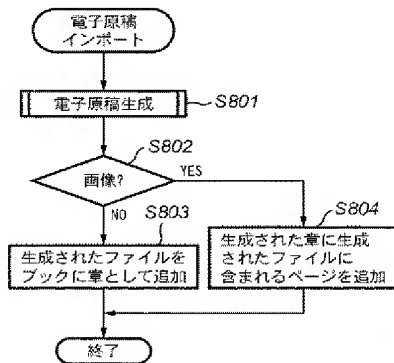
【図6】

| NO | 属性情報 | 備考 |
|----|----------|----------|
| 1 | ページ印刷指定 | ページ印刷指定 |
| 2 | ウォーダーマーク | ウォーダーマーク |
| 3 | ヘッダ・フッタ | ヘッダ・フッタ |
| 4 | ズーム | ズーム |
| 5 | 表紙位置 | 表紙位置 |
| 6 | インデックス | インデックス |
| 7 | 表紙位置 | 表紙位置 |
| 8 | ページ位置 | ページ位置 |

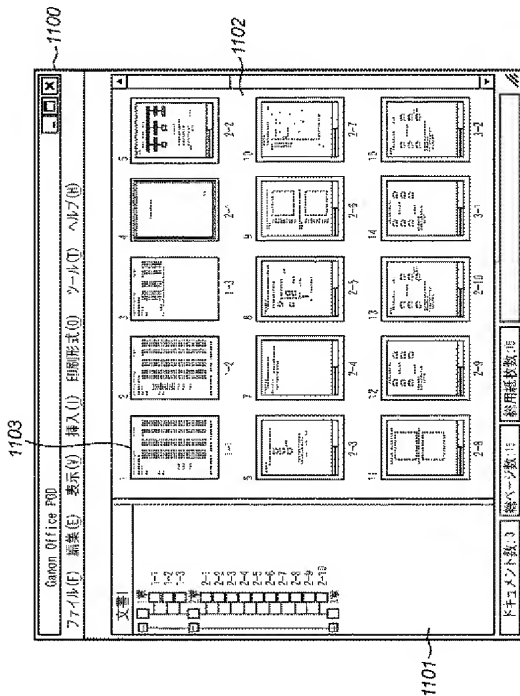
【図7】



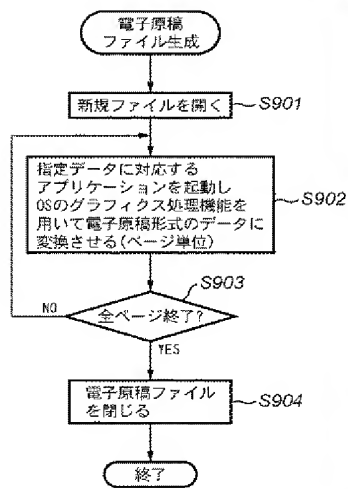
【図8】



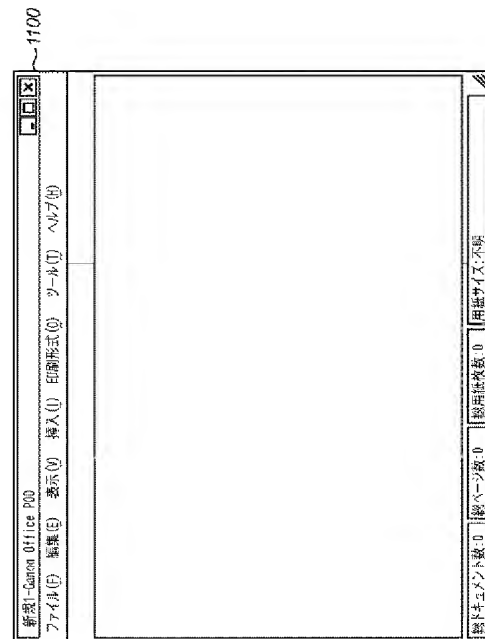
【図10】



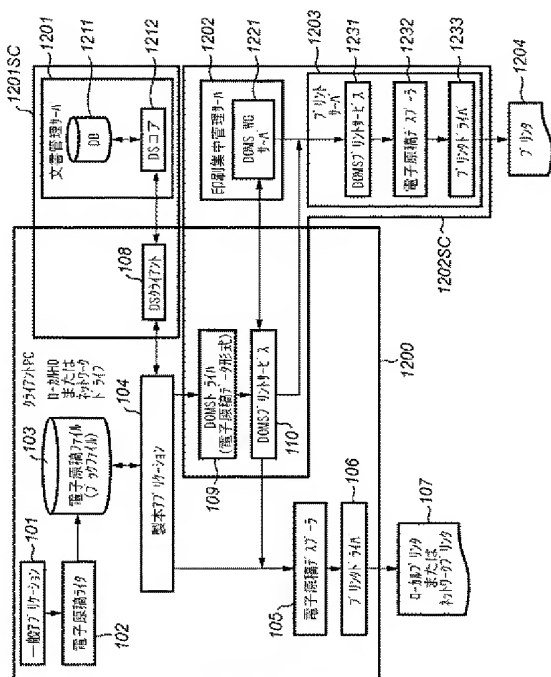
【図9】



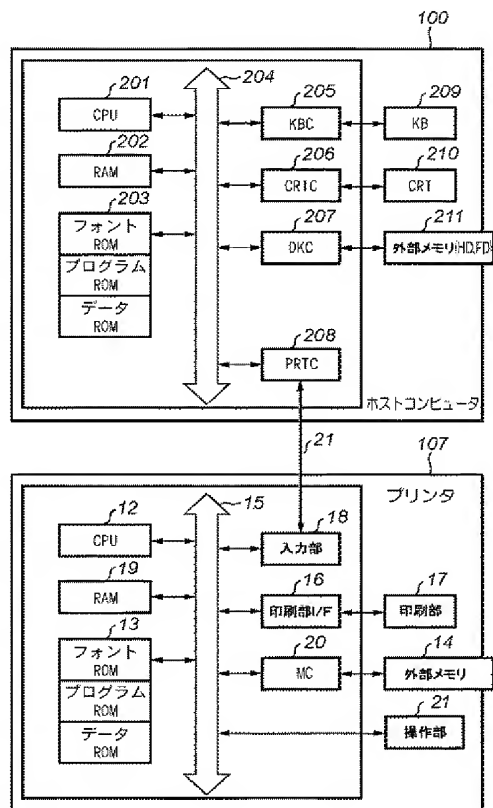
【図11】



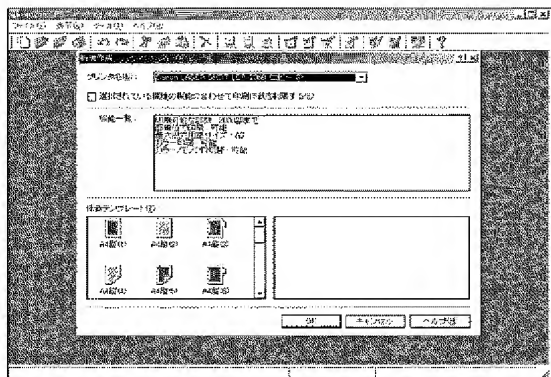
【図12】



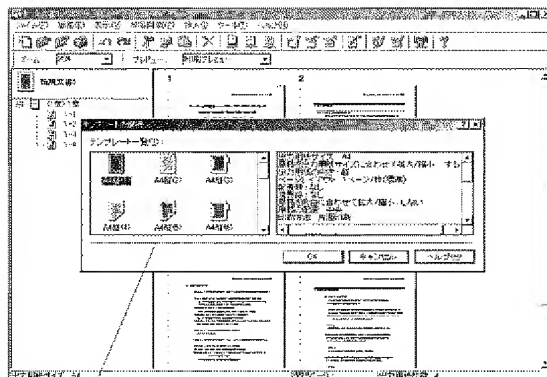
【図13】



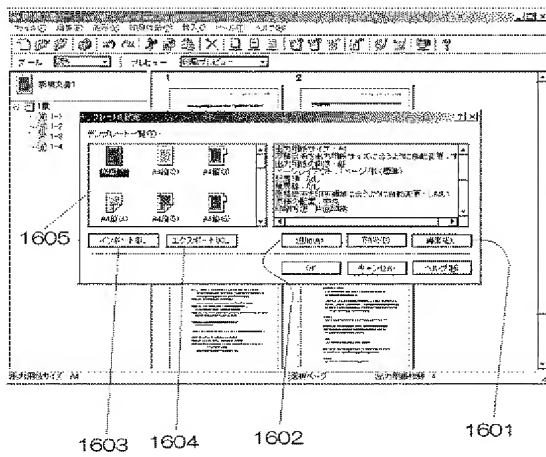
【図14】



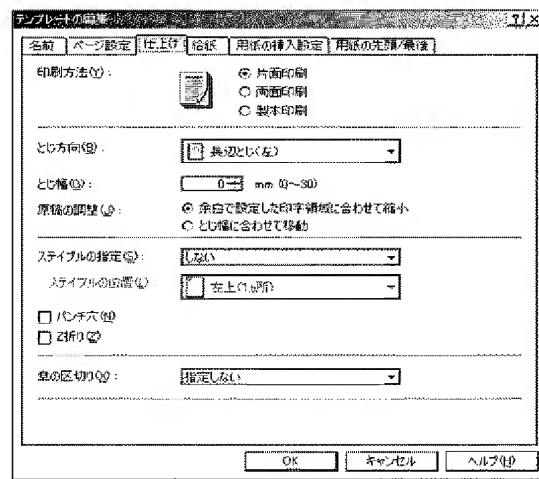
【図15】



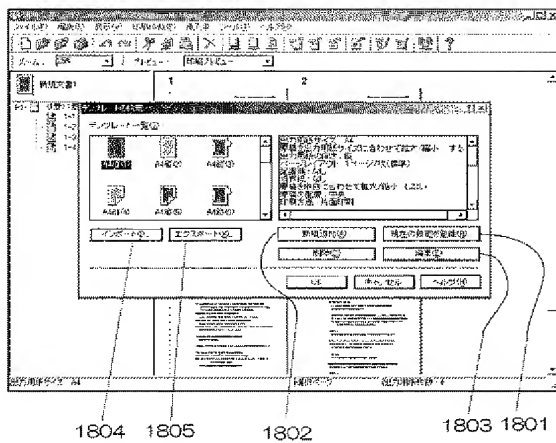
【図16】



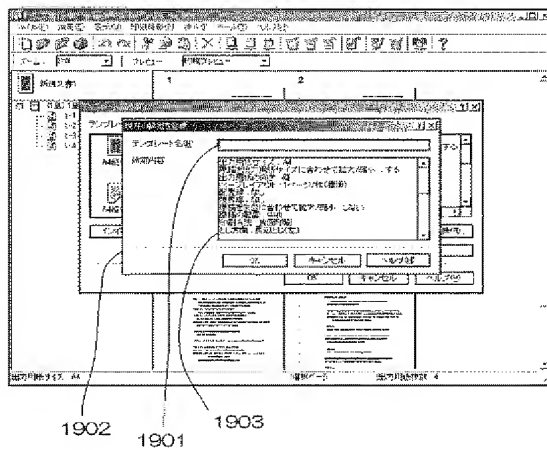
【図17】



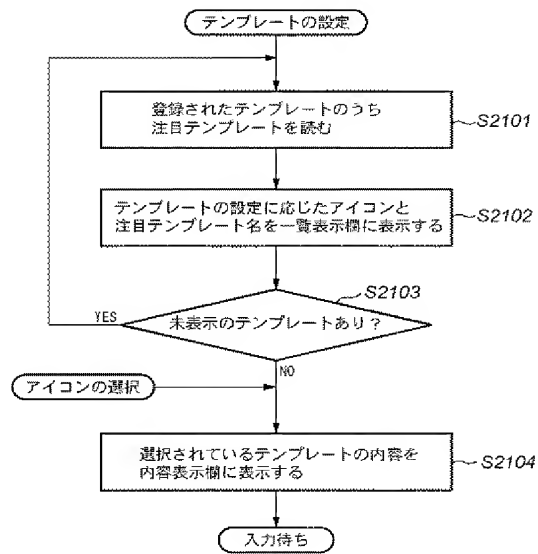
【図18】



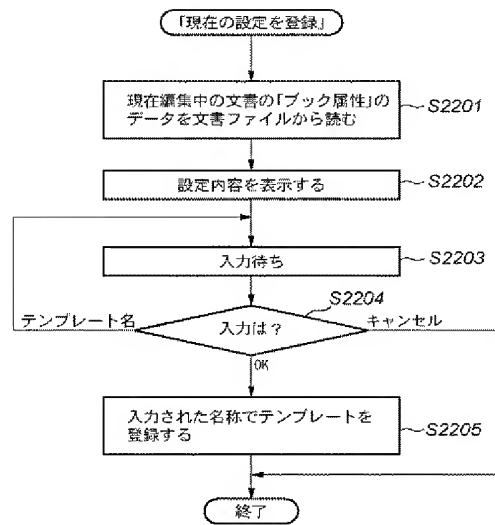
【図19】



【図20】



【図21】



(72)発明者 佐藤 純子

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

Fターム(参考) 5B009 NA03 NA07 RC01

5B021 AA01 LA01